

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-020428

(43)Date of publication of application : 21.01.2000

(51)Int.Cl.

G06F 13/00

G06F 15/16

H04L 12/28

H04L 12/24

H04L 12/26

H04L 29/14

(21)Application number : 10-191772

(71)Applicant : SUMITOMO ELECTRIC IND LTD

(22)Date of filing : 07.07.1998

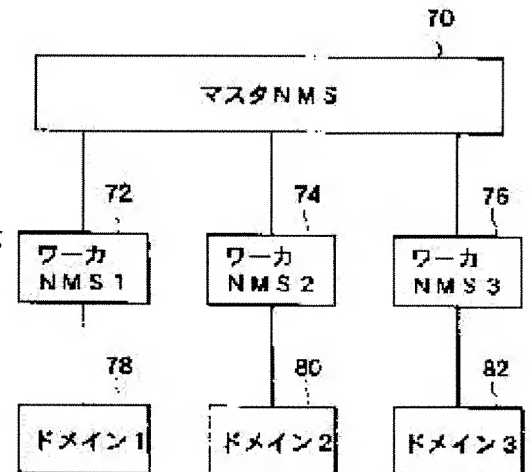
(72)Inventor : TAKENAMI YOSHINORI

## (54) NETWORK MANAGEMENT SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To effectively specify a problem in distributed environments by communicating propagation between the domains of symptom events also between respective network management systems together with the identifier(ID) of a problem event.

**SOLUTION:** A master network management system(NMS) 70 judges whether the ID of a problem event is requested or not, and when the ID is requested, transmits a PS matrix generation continuation request to other worker NMSs 72, 74, 76 for managing a boundary management object(MO) in response to the ID. The worker NMSs 74, 76 respectively use the request message as a trigger and pad a matrix by using the topology information of their corresponding domains 80, 82 but the worker NMS 72 is started from the problem of the MNSs 74, 76 and pads the matrix by using the topology information of its domain 78. These worker NMSs 72, 74, 76 communicate with each other.



(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2000-20428

(P2000-20428A)

(43)公開日 平成12年1月21日(2000.1.21)

| (51)Int.Cl. <sup>7</sup>             | 識別記号  | F I           | テーマコード*(参考)       |
|--------------------------------------|-------|---------------|-------------------|
| G 0 6 F 13/00                        | 3 5 1 | G 0 6 F 13/00 | 3 5 1 N 5 B 0 4 5 |
| 15/16                                | 4 6 0 | 15/16         | 4 6 0 Z 5 B 0 8 9 |
| H 0 4 L 12/28                        |       | H 0 4 L 11/00 | 3 1 0 D 5 K 0 3 0 |
| 12/24                                |       | 11/08         | 5 K 0 3 3         |
| 12/26                                |       | 13/00         | 3 1 3 5 K 0 3 5   |
| 審査請求 未請求 請求項の数 5 O L (全 20 頁) 最終頁に続く |       |               |                   |

(21)出願番号 特願平10-191772

(22)出願日 平成10年7月7日(1998.7.7)

(71)出願人 000002130

住友電気工業株式会社

大阪府大阪市中央区北浜四丁目5番33号

(72)発明者 武並 佳則

大阪市此花区島屋一丁目1番3号 住友電  
気工業株式会社大阪製作所内

(74)代理人 100064746

弁理士 深見 久郎 (外2名)

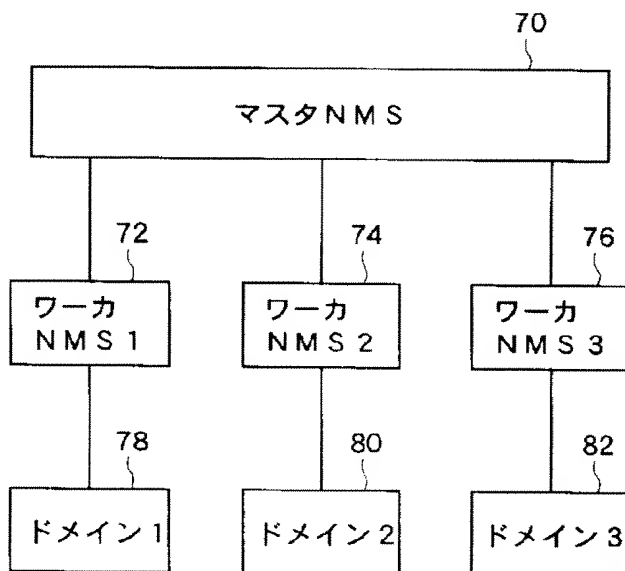
最終頁に続く

(54)【発明の名称】 ネットワーク管理システム

(57)【要約】

【課題】 分散環境で効果的に問題の特定を行うことができるネットワーク管理システム(NMS)を提供すること。

【解決手段】 ネットワーク管理システムは、ワーカNMS 72、74、76と、ワーカNMSの各々と接続されたマスタNMS 70とを含む。各ワーカNMSおよびマスタNMS 70は、相互に通信するためのNMSインタフェース部と、管理対象のドメインの構成情報データおよび障害情報データを取得するネットワークインタフェース部と、管理対象オブジェクトモデル、イベント伝播モデル、構成情報データ、問題イベントリスト、および他ドメインから伝播する症状イベント情報に基づき、自己のドメインでの症状イベントと、ネットワーク全体での問題との間の部分相関マトリクスを生成して送信する構成管理部とを含む。



**【特許請求の範囲】**

**【請求項1】** 自己の属するネットワークにおける問題を特定する際に参照される問題と症状パターンとの相関マトリクスを分散管理するためのネットワーク管理システムであって、

他のネットワーク管理システムと交信するための関係システム交信手段と、

管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段と、

管理対象オブジェクトモデルと、オブジェクトクラス間に定義されたイベントの伝播モデルと、前記データ取得手段により取得された構成情報データと、自己の管理するドメインでの管理の対象となる問題イベントリストと、前記関係システム交信手段を介して他ネットワーク管理システムから与えられる、他ドメインから伝播する症状イベント情報とに基づき、自己の管理するドメインでの症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成するための部分相関マトリクス生成手段と、

前記部分相関マトリクス生成手段によるマトリクス生成の過程で、他ドメインに伝播する症状イベントを検出し、当該症状イベントの原因となる問題イベントの識別子とともに、当該他ドメインを管理する他ネットワーク管理システムに対して前記関係システム交信手段を介して相関マトリクス生成継続要求を送信するための手段とを含む、ネットワーク管理システム。

**【請求項2】** さらに、前記データ取得手段により取得される、自己の管理するドメイン内で生じる症状イベントパターンと、前記部分相関マトリクスとの比較に基づいて、ネットワーク上で発生した問題イベントを特定し、所定の送信先に対して通知するための手段を含む、請求項1に記載のネットワーク管理システム。

**【請求項3】** 所定のネットワーク内で生ずる症状イベントから、ネットワーク障害の原因を推定するためのネットワーク管理システムであって、

複数個のワーカーネットワーク管理システムと、前記複数個のワーカーネットワーク管理システムの各々と接続されたマスタネットワーク管理システムとを含み、各前記ワーカーネットワーク管理システムは、

前記マスタネットワーク管理システムと交信するための交信手段と、

管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段と、

管理対象オブジェクトモデルと、オブジェクトクラス間に定義されたイベントの伝播モデルと、前記データ取得手段により取得された構成情報データと、自己の管理するドメインでの管理の対象となる問題イベントリストと、前記交信手段を介して前記マスタネットワーク管理システムから与えられる、他ドメインから伝播する症状イベント情報とに基づき、自己の管理するドメインでの

症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成するための部分相関マトリクス生成手段と、

前記部分相関マトリクス生成手段によるマトリクス生成の過程で、他ドメインに伝播する症状イベントを検出し、当該症状イベントの原因となる問題イベントの識別子と、当該他ドメインを管理するワーカーネットワーク管理システムの識別子とを含む相関マトリクス生成継続要求を、前記関係システム交信手段を介して前記マスタネットワーク管理システムに対して送信するための手段とを含み、

前記マスタネットワーク管理システムは、

各前記ワーカーネットワーク管理手段からの相関マトリクス生成継続要求を、識別子により特定されるワーカーネットワーク管理システムに送信するための手段を含む、ネットワーク管理システム。

**【請求項4】** 各前記ワーカーネットワーク管理システムは、前記部分相関マトリクス生成手段による部分相関マトリクスの生成の完了に応答して終了通知を前記マスタネットワーク管理システムに送信するための手段をさらに含み、

前記マスタネットワーク管理手段は、全ての前記ワーカーネットワーク管理システムから終了通知を受信したことに応答して、各前記ワーカーネットワーク管理システムに対して相関処理の開始を指令するための手段をさらに含み、

各前記ワーカーネットワーク管理システムは、前記指令に応答して、自己の管理するドメイン内で生じる症状イベントパターンと、前記部分相関マトリクスとの比較を行う処理を開始し、自己の管理するドメイン内で発生した可能性の高い問題イベントを特定し、前記マスタネットワーク管理システムに対して通知するための手段をさらに含み、

前記マスタネットワーク管理システムは、各前記ワーカーネットワーク管理システムからの、発生した可能性の高い問題イベントについての通知に基づいて前記ネットワークの障害の可能な根本原因を特定するための手段をさらに含む、請求項3に記載のネットワーク管理システム。

**【請求項5】** 所定のネットワーク内で生ずる症状イベントから、ネットワーク障害の原因を推定するためのネットワーク管理システムであって、

複数個のワーカーネットワーク管理システムと、

前記複数個のワーカーネットワーク管理システムの各々と接続されたマスタネットワーク管理システムとを含み、各前記ワーカーネットワーク管理システムは、

前記マスタネットワーク管理システムと交信するための交信手段と、

管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段とを含む、

前記マスタネットワーク管理システムは、各前記ワーカネットワーク管理手段からネットワークの管理対象オブジェクト識別名と、管理対象オブジェクトクラス名と、管理対象オブジェクト間の接続関係情報とを取得して、ネットワークにおける問題イベントと症状イベントとの相関マトリクスを生成するための手段と、前記相関マトリクスを各前記ワーカネットワーク管理システムごとに、各前記ワーカネットワーク管理システムが管理するドメインでの発生が想定される症状イベントに基づいて分割した部分相関マトリクスを当該ワーカネットワーク管理システムに送信するための手段とを含み、

各前記ワーカネットワーク管理システムは前記部分相関マトリクスを保持するための手段と、

自己の管理するドメインの症状イベントパターンと前記部分相関マトリクスとを比較することにより、ネットワーク内の障害の原因である可能性の高い問題イベントを特定して前記マスタネットワーク管理システムに通知するための手段とを含み、

前記マスタネットワーク管理システムは、各前記ワーカネットワーク管理システムからの、発生した可能性の高い問題イベントについての通知に基づいて前記ネットワークの障害の可能な根本原因を特定するための手段をさらに含むネットワーク管理システム。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】本発明は、ネットワーク上の障害を管理するネットワーク管理システムに関し、特に、ネットワーク上で観測されるさまざまな複数の障害の症状から障害の根本原因を特定する機能を有するネットワーク管理システムに関する。

##### 【0002】

【従来の技術】コンピュータによる通信ネットワークの大規模化が進んでいる。通信ネットワークが大規模化するに従って、ネットワーク上に発生する障害の及ぼす影響も大規模かつ深刻なものとなりつつある。そのためネットワーク管理をいかに効率よく行なうか、が非常に重要である。以下、本明細書上で使用されるネットワーク管理に関する用語について定義をする。

【0003】「イベント」とは、ネットワークにおいて発生する例外的な状態のことをいう。「イベント」という概念はハードウェアやソフトウェアの故障、停止、性能のボトルネック、ネットワークの構成の不整合、設計不十分による意図せざる結果、コンピュータウィルス等の悪意による被害などを含む。

【0004】「症状」とは、観測可能なイベントのことをいう。「症状イベント」と同じである。たとえば「ある宛先Aに対して常に通信に時間がかかり再送信が必要となる」、「ある宛先Bに対していつも文字化けが生ずる」、「ある宛先Cに対していつも受信確認が返ってこ

ない」などの事象をいう。同じ意味で「Symptom」という語も使用する。

【0005】「問題」とは、障害の根本原因のことをいう。必ずしも観測可能ではない。たとえば通信装置の送信機破損、通信ケーブルの断線、通信回線の容量不足などが例である。「問題イベント」は「問題」と同じ意味である。同じ意味で「プロBLEM」(Problem)という語も使用する。

【0006】「オブジェクト」とは、概念や抽象または対象となる問題に対して明確な境界と意味を持つ何ものか、のことをいう。「オブジェクトインスタンス」とは、後述するあるオブジェクトクラスに属するある特定の1つのオブジェクトのことをいう。単に「インスタンス」ともいう。「オブジェクトクラス」とは、同様の性質(属性)、共通の振る舞い(操作)、他のオブジェクトとの共通の関係、および共通の意味を持つオブジェクトのグループをいう。「クラス」はオブジェクトクラスと同じである。「サブクラス」とは、あるクラスの下位クラスとして、そのクラスに包含されるクラスのことをいう。「属性」とは、クラスに属する各オブジェクトによって保持されるデータをいう。

【0007】「オブジェクト図」とは、オブジェクト、クラス、それらの間の関係のモデル化のための、形式的な図式記法をいう。「クラス図」とは、多くの可能なインスタンスを記述するためのスキーマ、パターン、テンプレートなどである。「リポジトリ」とは、必要な情報を集約して一覧表的な形式で記憶した記憶部を言う。集約一覧表メモリとも呼ぶべきものである。「イベントリポジトリ」とは、実際に発生した症状パターンを記憶する記憶部をいう。「オブジェクトリポジトリ」とは、ネットワークの構成モデルのリポジトリのことをいう。

【0008】ネットワークのあるリソースにおける1つの問題イベントは、関係する複数のリソースの多くの症状イベントを引き起こし得る。問題の中には、観測可能なイベントであるものもあるが、一般には必ずしも観測可能ではない。そのため複数の症状から障害の根本原因である問題を特定する必要がある。したがって、ネットワーク管理者は、根本原因の問題を特定するために、観測される種々の症状イベントを問題と相関させることができればならない。

【0009】しかし、ネットワークが大規模になると、観測される症状イベントの数も膨大になる。またどの問題がどの症状を引き起こすかという「因果関係」とでも言うべきものも複雑になってくるために、オペレータが手作業で障害の根本原因の問題を特定することはほとんど不可能となる。

【0010】このようなネットワーク上で観測される膨大な障害の症状イベントから根本原因の問題を正確にかつ高速に特定するための従来技術手法として、1996年6月18日発行の米国特許第5,528,516号

(「Apparatus and Method for Event Correlation and Problem Reporting (イベント相関および問題報告装置および方法)」)が提案されている。

【0011】この従来技術は次の2つの技術に分けることができる。

- (1) 管理対象ネットワークのモデリング技術
- (2) イベント相関技術

前者の「管理対象ネットワークのモデリング技術」とは、実際のネットワーク上で発生する問題イベントおよび症状イベントをいかに正確に効率よくモデル化するかということに関する。後者のイベント相関技術とは、主として、観測される膨大な症状イベントからいかに高速に根本原因である問題を特定するか、ということに関する。本発明は(2)のイベント相関技術に関連する。そのため以下では(1)の管理対象ネットワークのモデリング技術については、本願発明の理解の上で必要な最小限の説明にとどめる。なお以下の説明は、障害に関するイベントに限定して行なうが、イベントの種類はどのようなものであってもよく、本発明は障害に関するイベントのみに制約されるものではない。

【0012】この従来技術で提案されている「管理対象ネットワークのモデリング技術」について以下に簡単に説明する。まず、ルータおよびハブなどのネットワーク機器と、パーソナルコンピュータおよびワークステーションなどのコンピュータと、これらを接続するネットワークそのものと、コンピュータ等の上で実行されるソフトウェア等とからなるすべてのものを管理対象オブジェクトManagedObject (MO)としてモデル化する。そして、管理対象オブジェクト間の「関係」を重要視し、ネットワーク上で発生する問題イベントおよび症状イベントは、管理対象オブジェクト間に設定された関係に沿って「伝播」するものとしてイベントの伝播をモデル化する。

【0013】上述の従来技術では、このような管理対象オブジェクトのモデル化およびイベント伝播のモデル化の静的な側面を抽象化し、モデル化を効率的に行なうためにオブジェクト指向の概念を導入している。すなわち種々の管理対象オブジェクトをクラスとしてモデル化する。そしてクラス間の関係を定義する。さらにあるイベントは、クラス間の関係に沿って伝播するものとしてモデル化される。オブジェクト指向技術については種々の教科書があるのでそれらを参照されたい。

【0014】まず、管理対象オブジェクトクラスのモデル化について説明する。図11に、上述の従来技術で提案されている管理対象オブジェクトモデル(クラス階層)を示す。図11および他の図面において、管理対象オブジェクトモデルについてはOMT記法を用いて記す。図11を参照して、各ボックスはオブジェクトクラスを示す。オブジェクトクラスの各ボックスは2つのボックスに分割されている。上部のボックスに記載されて

いるのはオブジェクトクラス名であり、下部のボックスに記載されているのはそのクラスのオブジェクトが持つべき属性である。各クラス間には関係が設定され、各関係はクラス間に引かれた線で示されている。各線の近傍には、その関係の名称が記載されている。

【0015】図11に示されるモデルは、ManagedObject クラス110と、Element クラス112と、Layer クラス114と、Group クラス116と、Resourceクラス118と、PhysicalService クラス120と、LogicalServiceクラス122と、Nodeクラス124と、Linkクラス126と、Manager クラス128を含む。Manager クラス128がネットワーク管理システムのクラスである。

【0016】これらクラス間に設定される関係は、その関係により結びつけられる2つのクラスのいずれから見るかによって名称が変わる。たとえばManager クラス128とResourceクラス118との関係は、Manager クラス128から見ればManages、Resourceクラス118から見ればManaged-byとなる。またたとえばNodeクラス124とLinkクラス126との間の関係208は一方から見ればConnecte-d-via、他方から見ればConnected-toである。またたとえばElement クラス112から出て同じクラス内に戻る関係206(Layered-overとUnderlying)のように、同一クラス内のオブジェクト間に定められる関係もある。

【0017】こうして定められたクラスシステムにもとづいて、管理対象のネットワークをモデル化する。すなわち、ネットワーク内の管理対象オブジェクトをあるクラスの一つのインスタンスとして抽象化し、そのインスタンスが属するクラスと、他のインスタンスが属するクラスとの間に設定された関係にしたがってイベントがこれらインスタンス(管理対象オブジェクト)を伝播していくものとしてネットワークをモデル化する。さらに、こうしてモデル化されたネットワークに基づき、問題と、症状との間の相関を予め特定する。

【0018】まず、管理対象オブジェクトクラスのモデル化について説明する。今、実際のネットワークの例として図12に示されるネットワークを考える。このネットワークは、互いにブリッジ140によって結合された2つのイーサネットワーク134およびイーサネットワーク136を含む。一方のイーサネットワーク134にはホスト130が、他方のイーサネットワーク136にはホスト132がそれぞれ接続されているものとする。以下の例ではホスト130にネットワーク管理システム(NMS)が搭載されているものとして考える。

【0019】図12に示されるネットワークを上述した考え方をを用いてモデル化するために次のようにする。図13を参照して、各ボックスはオブジェクトクラス(またはサブクラス)を示す。オブジェクトクラスの各ボックスは2つのボックスに分割されている。上部のボックスに記載されているのはオブジェクトクラス名であり、下部のボックスに記載されているのはそのクラスのオブ

ジェクトが持つべき属性である。各クラス間には関係が設定され、各関係はクラス間に引かれた線で示されている。各線の近傍には、その関係の名称が記載されている。なお図13において、管理対象オブジェクトモデルについてはOMT記法を用いて記す。

【0020】図13に示されるように、Nodeクラス124の下に3つのサブクラスTcpNode クラス154、IpNodeクラス156 およびEtherNode クラス158 を定義する。さらにLinkクラス126 のサブクラスとしてTcpLink クラス160、IpLinkクラス162、EtherLink クラス164 およびMacBridgeLink クラス166 を定義する。なお、各クラスにはそのクラス特有の属性を定義する必要がある。たとえばIpNodeクラスにはIP(Internet Protocol) アドレス属性を定義するなど、である。しかし、本発明には属性の定義は関係しないので、ここでは属性についての詳細な説明は行わない。

【0021】これらクラス間に設定される関係は、その関係により結びつけられる2つのクラスのいずれから見るかによって名称が変わる。たとえばNodeクラス124 とLinkクラス126 との間の関係208 は一方から見ればConnected-via、他方から見ればConnected-toである。また図13には示していないが、同一クラス内のオブジェクト間に定められる関係もあり得る。

【0022】図13に示されるクラスを用いて、図12に示すネットワーク例をインスタンス間の関係図として表現することができる。こうした図は、オブジェクト指向技術では「インスタンス図」と呼ばれる。

【0023】図14を参照して、ホスト130には、それぞれManager クラス、TcpNode クラス、IpNodeクラスおよびEtherNode クラスのインスタンスであるインスタンス170、172、174 および176 が含まれる。一方ホスト132には、TcpNode クラス、IpNodeクラスおよびEtherNode クラスのインスタンスであるインスタンス182、184 および186 が含まれる。インスタンス172 および182はTcpLink クラスのインスタンス200によって結合される。インスタンス174 および184はIpLinkのインスタンスであるインスタンス202によって結合される。

【0024】ブリッジ140は、EtherNode クラス、MacBridgeNode クラスおよびEtherNodeクラスのインスタンスであるインスタンス190、192 および194 を含んでいる。インスタンス176 および190はEtherLink クラスのインスタンスであるインスタンス196により結合される。インスタンス194 および186は同じくEtherLink クラスのインスタンス198により結合される。インスタンス190、192、194、196 および198はそれぞれインスタンス202と結合されている。

【0025】このようにしてモデル化された管理対象オブジェクトに対して、症状イベントの伝播ルールが予め準備される。この伝播ルールは、障害の根本原因の問題イベントが障害の症状イベントに伝播し、その症状イベ

ントが別の症状イベントに伝播するという関係をルール化したものである。この伝播ルールの集合を伝播モデルと呼ぶ。伝播モデルの例を以下に示す。

【0026】・EtherLink がDownすれば、Connected-to 関係に従ってEtherNode がDownする。

【0027】・EtherNode がDownすれば、Underlying関係に沿ってIpNodeがDownする。

・EtherNode がDownすれば、Connected-via 関係に沿ってMacBridgeLink がCannotBridgingである。

【0028】・IpNodeがDownすれば、Underlying関係に沿ってTcpNode がDisconnectする。

・TcpNode がDisconnectすれば、Connected-via 関係に沿ってTcpLink がNotResponding である。

【0029】・TcpLink がNotResponding であると、Connected-to関係に沿ってそれに接続されるTcpNode がDisconnectである。

【0030】上記したルールにおいて、Down、DisconnectおよびNotResponding が障害の問題イベントおよび／または症状イベントである。イベントの中には問題イベントでかつ症状イベントであるものもあるし、どちらでもないものもある。このような伝播モデル(ルール)を図式化して図15に示す。図15において、上記したイベントは、管理対象オブジェクトのクラス間に定義されている関係に沿って伝播するという、イベント伝播のモデル化がなされている。図15に示される各クラス154、156、158、160、164 および166の間では、それぞれの関係210、212、214、216、218 および220 という関係(順不同)があり、上記したルールに従ってこれらの関係に沿ってイベントが伝播する。

【0031】図15に示されるイベント伝播モデルが想定されている場合に、図12に示され図14にモデル化されているネットワークでイベントがどのように伝播するかを以下に示す。ここでは障害の根本原因の問題としてEtherLink1がDown(これを「ケース1」と称する)とEtherLink2がDown(これを「ケース2」と称する)の場合を考える。この場合、イベント伝播モデルに従って観測されると記載される症状イベントは図16の表1に示されるようになる。

【0032】これらの表において、最も左側の列の各欄には症状イベントが、第1行目の各欄には問題イベントが、それぞれ記載されている。ある問題イベントの列とある症状イベントの行との交わる欄には、その問題イベント(原因)が発生したときに、当該症状イベントが発生するか否かを示す。この場合症状イベントが生ずる場合に「1」を記入するものとする。たとえばEtherLink1がDownが発生したときにはEcpNode1.Disconnect が生じるが、EtherNode2.Down は生じない。なお各インスタンス名は図14に記載のものをを用いる。

【0033】各ケースについて該当の列を縦に見ていくと、「1」および「空白(0)」を連ねたものが得られ

る。これを当該問題イベントに対する症状イベント群のコードと称する。たとえば表1(図16)のケース1では症状イベント群のコードは「1101101101」である。未記入(空白)の箇所は0としてある。

【0034】図16に示される表1からわかるように、問題イベントが異なると、観測される症状イベント群のコードが異なる。したがって実際にネットワーク管理システム(NMS)により観測および収集される症状イベント群と、図16の表1に示される症状イベント群とを比較することにより、障害の根本原因の問題を特定することが可能となる。このように、各問題と症状イベントとの相関関係を表したものをPSマトリクスまたは問題と症状イベント群との相関マトリクスと呼ぶ。実際にはこの比較は、得られた症状イベント群と、PSマトリクスの症状イベント群との間に定義される「距離」(または類似の尺度)を計算し、最も距離の小さい問題(または最も類似の尺度の大きな問題)を特定することにより行なわれる。

【0035】一般に、管理対象ネットワークが巨大となり、管理対象オブジェクト数が増え、それに伴って問題イベント数および症状イベント数が膨大な数となる。そのためPSマトリクスは巨大なマトリクスとなる。すると、メモリの都合などによって1台のネットワーク管理システムでは巨大なネットワークを集中管理することが困難となり、複数のネットワーク管理システムで管理対象ネットワークを分散管理する必要が生じる。

【0036】従来技術では、分散環境でのイベント相関を以下のようにして解決しようとしている。すなわち、あるドメインのモデルを考えると、他のドメインについては、当該他のドメインを代表する特別な管理対象オブジェクトとして表現する。そして各ドメインごとにPSマトリクスを生成してドメインごとに個別にイベント相関を行う。従来技術に関する文献では、enterpriseドメインとrouterドメインとが設けられ、enterpriseドメインの観点からはrouterドメインはrouter backboneオブジェクトを表すものとして、Linkの特別な場合としてモデル化している。そして、一方のドメインでのイベントが他方のドメインでのイベントの症状イベントである可能性がある、そのイベントを症状イベントとして他方のドメインに通知し、当該他のドメインが、通知された症状イベントをも考慮してPSマトリクスを用いてイベント相関を行う。各ドメインでのPSマトリクスの形式の例を図17に示す。

【0037】

【発明が解決しようとする課題】このようにある問題イベントに対して生ずる症状イベントを予めパターン化しておけば、実際に障害が生じたときの症状パターンとこのパターンとを比較するという比較的単純な作業により障害の根本原因の問題を特定することができる。したがって、この従来の技術により障害の根本原因の問題の特

定が非常に容易になるかと思われる。しかしこの従来の技術には次のような問題点がある。

【0038】上記した従来技術のようなPSマトリクスを用いる方法では、ドメイン間の管理対象オブジェクト(境界管理対象オブジェクト)でイベント伝播が集約される。そのため、複数のドメインにまたがるクロスドメイン全体としては正確なイベント相関ができない可能性がある。また、境界管理対象オブジェクトを、分散環境を考慮した特別な管理対象オブジェクトとしてモデル化する必要がある。さらに、個々のドメインでPSマトリクスを形成して個々のドメインで生ずるイベントと問題との相関付けを行うため、あるドメインでの症状イベントと、他のドメインでの問題イベントとの間の関連付けをするための情報が基本的に欠落するという問題がある。

【0039】それゆえに本願発明の目的は、分散環境で効果的に問題の特定を行うことができるネットワーク管理システムを提供することである。

【0040】本願発明の他の目的は、分散環境で効果的に問題の特定を行うためのPSマトリクスを作成することができるネットワーク管理システムを提供することである。

【0041】本願発明の他の目的は、分散環境で効果的に問題の特定を行うためのPSマトリクスを用いてクロスドメインでの問題特定を行うことができるネットワーク管理システムを提供することである。

【0042】

【課題を解決するための手段】請求項1に記載の発明にかかるネットワーク管理システムは、自己の属するネットワークにおける問題を特定する際に参照される問題と症状パターンとの相関マトリクスを分散管理するためのネットワーク管理システムである。このシステムは、他のネットワーク管理システムと通信するための関係システム通信手段と、管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段と、管理対象オブジェクトモデルと、オブジェクトクラス間に定義されたイベントの伝播モデルと、データ取得手段により取得された構成情報データと、自己の管理するドメインでの管理の対象となる問題イベントリストと、関係システム通信手段を介して他ネットワーク管理システムから与えられる、他ドメインから伝播する症状イベント情報とに基づき、自己の管理するドメインでの症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成するための部分相関マトリクス生成手段と、部分相関マトリクス生成手段によるマトリクス生成の過程で、他ドメインに伝播する症状イベントを検出し、当該症状イベントの原因となる問題イベントの識別子とともに、当該他ドメインを管理する他ネットワーク管理システムに対して関係システム通信手段を介して相関マトリクス生成継続



要求を送信するための手段とを含む。

【0043】症状イベントのドメイン間での伝播を問題イベントの識別子とともに各ネットワーク管理システムの間で交信するので、各ネットワーク管理システムでは、自己の管理するドメインでの症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成することができる。この部分相関マトリクスには、従来と異なり問題イベントと症状イベントとのドメインが異なったときの情報の欠落がない。したがって、この部分相関マトリクスを用いて、従来の方式よりも正確な問題の推定を行うことができる。

【0044】請求項2に記載の発明にかかるネットワーク管理システムは、請求項1に記載の発明の構成に加えて、さらに、データ取得手段により取得される、自己の管理するドメイン内で生じる症状イベントパターンと、部分相関マトリクスとの比較に基づいて、ネットワーク上で発生した問題イベントを特定し、所定の送信先に対して通知するための手段を含む。

【0045】各ネットワーク管理システムでの部分相関マトリクスによる問題の特定の結果を所定の送信先に送信することにより、当該送信先では、複数個のドメインからの問題の特定結果を参照することができ、これらのうちから最も可能性の高い問題を根本原因として特定することができる。

【0046】請求項3に記載の発明にかかるネットワーク管理システムは、所定のネットワーク内で生ずる症状イベントから、ネットワーク障害の原因を推定するためのネットワーク管理システムであって、複数個のワーカネットワーク管理システムと、これら複数個のワーカネットワーク管理システムの各々と接続されたマスタネットワーク管理システムとを含む。各ワーカネットワーク管理システムは、マスタネットワーク管理システムと交信するための交信手段と、管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段と、管理対象オブジェクトモデルと、オブジェクトクラス間に定義されたイベントの伝播モデルと、データ取得手段により取得された構成情報データと、自己の管理するドメインでの管理の対象となる問題イベントリストと、交信手段を介してマスタネットワーク管理システムから与えられる、他ドメインから伝播する症状イベント情報とに基づき、自己の管理するドメインでの症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成するための部分相関マトリクス生成手段と、部分相関マトリクス生成手段によるマトリクス生成の過程で、他ドメインに伝播する症状イベントを検出し、当該症状イベントの原因となる問題イベントの識別子と、当該他ドメインを管理するワーカネットワーク管理システムの識別子とを含む相関マトリクス生成継続要求を、関係システム交信

手段を介してマスタネットワーク管理システムに対して送信するための手段とを含む。マスタネットワーク管理システムは、各ワーカネットワーク管理手段からの相関マトリクス生成継続要求を、識別子により特定されるワーカネットワーク管理システムに送信するための手段を含む。

【0047】症状イベントのドメイン間での伝播をマスタネットワーク管理システムを介して各ワーカネットワーク管理システムの間で交信するので、各ワーカネットワーク管理システムでは、自己の管理するドメインでの症状イベントと、ネットワーク全体での管理の対象となる問題イベントとの間の部分相関マトリクスを生成することができる。この部分相関マトリクスには、従来と異なり問題イベントと症状イベントとのドメインが異なったときの情報の欠落がない。したがって、この部分相関マトリクスを用いて、従来の方式よりも正確な問題の推定を行うことができる。

【0048】請求項4に記載の発明にかかるネットワーク管理システムは、請求項3に記載のシステムであって、各ワーカネットワーク管理システムは、部分相関マトリクス生成手段による部分相関マトリクスの生成の完了にตอบสนองして終了通知をマスタネットワーク管理システムに送信するための手段をさらに含む。マスタネットワーク管理手段は、全てのワーカネットワーク管理システムから終了通知を受信したことにตอบสนองして、各ワーカネットワーク管理システムに対して相関処理の開始を指令するための手段をさらに含む。各ワーカネットワーク管理システムは、この指令にตอบสนองして、自己の管理するドメイン内で生じる症状イベントパターンと、部分相関マトリクスとの比較を行う処理を開始し、自己の管理するドメイン内で発生した可能性の高い問題イベントを特定し、マスタネットワーク管理システムに対して通知するための手段をさらに含む。そしてマスタネットワーク管理システムは、各ワーカネットワーク管理システムからの、発生した可能性の高い問題イベントについての通知に基づいてネットワークの障害の可能な根本原因を特定するための手段をさらに含む。

【0049】各ワーカネットワーク管理システムでの部分相関マトリクスによる問題の特定の結果をマスタネットワーク管理システムに送信することにより、マスタネットワーク管理システムでは、複数個のドメインからの問題の特定結果を参照することができ、これらのうちから最も可能性の高い問題を根本原因として特定することができる。

【0050】請求項5に記載の発明にかかるネットワーク管理システムは、所定のネットワーク内で生ずる症状イベントから、ネットワーク障害の原因を推定するためのネットワーク管理システムであって、複数個のワーカネットワーク管理システムと、これら複数個のワーカネットワーク管理システムの各々と接続されたマスタネッ



トワーク管理システムとを含む。各ワーカネットワーク管理システムは、マスタネットワーク管理システムと通信するための通信手段と、管理対象のドメインの構成情報データおよび障害情報データを取得するためのデータ取得手段とを含む。マスタネットワーク管理システムは、各ワーカネットワーク管理手段からネットワークの管理対象オブジェクト識別名と、管理対象オブジェクトクラス名と、管理対象オブジェクト間の接続関係情報とを取得して、ネットワークにおける問題イベントと症状イベントとの相関マトリクスを生成するための手段と、当該相関マトリクスを各ワーカネットワーク管理システムごとに、各ワーカネットワーク管理システムが管理するドメインでの発生が想定される症状イベントに基づいて分割した部分相関マトリクスを当該ワーカネットワーク管理システムに送信するための手段とを含む。各ワーカネットワーク管理システムは前記部分相関マトリクスを保持するための手段と、自己の管理するドメインの症状イベントパターンと部分相関マトリクスとを比較することにより、ネットワーク内の障害の原因である可能性の高い問題イベントを特定してマスタネットワーク管理システムに通知するための手段とを含む。マスタネットワーク管理システムは、各ワーカネットワーク管理システムからの、発生した可能性の高い問題イベントについての通知に基づいてネットワークの障害の可能な根本原因を特定するための手段をさらに含む。

【0051】請求項5に記載の発明によれば、管理対象のネットワーク全体での相関マトリクスをマスタネットワークで生成し、これを分割した部分相関マトリクスを各ワーカネットワーク管理システムに送信する。この部分相関マトリクスには、従来と異なり問題イベントと症状イベントとのドメインが異なったときの情報の欠落がない。したがって、この部分相関マトリクスを用いて、各ワーカネットワーク管理システムで問題の特定を行う際に、従来の方式よりも正確な問題の推定を行うことができる。

#### 【0052】

【発明の実施の形態】上記した問題点の原因は、各ドメインでPSマトリクスを生成し、個別に問題イベント特定のための処理をすることにある。本願発明は、この従来技術の問題点を解決するために、クロスドメイン全体として1つのPSマトリクスを生成し、クロスドメイン全体として根本の原因である問題イベントを特定するために、データを分散管理し、処理を分散させることとした。なお「クロスドメイン」とは、以下に述べるようにネットワーク全体を複数のドメインに分けた場合の、複数ドメインにわたる領域のことをいう。また以下の説明でも、「MO」とは管理対象オブジェクトモデルのことを指す。

【0053】こうした処理を行うための枠組みを、本願発明では以下のようにして実現した。まず、イベント相

関処理について考えると、イベント相関処理は以下の二つの処理に分けることができる。

【0054】(1) PSマトリクスの生成

(2) 観測SymptomSet相関処理

SymptomSetとは、実際に観測されるSymptomの集合であって、SymptomSet[S11', S12', ..., S1n1', S21', S22', ..., S2n2', S31', S32', ..., S3n3']のように表現することとする。相関処理では、入力SymptomSetとPSマトリクスの各列との距離を求めて、その距離が最小となる問題Pを特定する。

【0055】<第1の実施の形態>

【システム構成】図1を参照して、本願発明に係るネットワーク管理システムで用いるPSマトリクス90は、図17に示される従来のものと異なり、全体として一つのPSマトリクス90として管理される。このPSマトリクス90を、全体としてPSマトリクス90を構成するようにデータを分散させて管理する。以下の実施の形態のシステムではPSマトリクス90を横方向の部分マトリクス92、94、96に分割して、別々のドメインで管理する。なお、本実施の形態のシステムでは管理対象ネットワークは図1にも示されるように3つのドメインA、B、Cに分割されるものとする。図1に示すPSマトリクス90では、従来技術では欠落していた領域[A2]、[A3]、[B1]、[B3]、[C1]、[C2]の部分の情報も相関処理で利用することができる。なお、ドメイン数が3に限定されないことはもちろんである。

【0056】PSマトリクス90を分散管理するためには、PSマトリクス90を図1における縦方向に分割することも考えられる。ただし、横方向に分割すると、各ドメインを管理するNMSにおいて貯えられた入力症状イベントの集合を、他のドメインのNMSに移動させる必要がないのでより好ましい。

【0057】図2を参照して、本実施の形態のシステムでは、上述した分散処理を実現するためにマスタ・ワーカ・モデルを採用した。すなわち、本ネットワーク管理システムは、第1のドメイン78を管理するためのワーカNMS72と、第2のドメイン80を管理するためのワーカNMS74と、第3のドメイン82を管理するためのワーカNMS76と、これらワーカNMS72、74、76を統括するマスタNMS70とを含む。

【0058】マスタNMS70のカバーする対象ドメインは、

(1) 統合ドメイン

(2) イベント相関に関係する複数のドメインの集合

(3) イベント相関に関係のない管理対象オブジェクトを予め除外できる場合は、管理対象オブジェクトのクラス名、ある属性の値の関係(等しい/大小)などの条件によりそれらの管理対象オブジェクトを除いて絞り込んだクロスドメインのドメイン

などが考えられる。

【0059】各ワーカNMS72、74、76は、図1に示す部分マトリクス92、94、96をそれぞれ管理する。これら部分マトリクス92、94、96の作成方法については後述する。

【0060】図3を参照して、本願発明マスタNMS70、ワーカNMS72、74、76などを構成するネットワーク管理システム20は、管理対象ネットワーク22に接続され、SNMP (Simple Network Management Protocol) 等のネットワーク管理プロトコルを用いて管理対象装置のMIB (Management Information Base) 等のデータを取得したり、ポーリングしたり、管理対象装置からのTrapイベントを受信したりするためのネットワーク管理インターフェイス部24と、管理対象オブジェクトモデルと、これに関連したイベント伝播モデルとを保持する機能と、これらモデルとネットワークの構成情報とに基づき管理対象オブジェクト構成情報を構築する機能と、自己の保持する管理対象オブジェクトモデルと、伝播モデルと、管理対象オブジェクトとを用いてPSマトリクスを構築するための構成管理部30と、構成管理部30により構築された管理対象オブジェクト構成情報を保持するための管理対象オブジェクトリポジトリ32と、構成情報データおよび障害情報データをイベントとして保持するためのイベントリポジトリ28と、ネットワーク管理インターフェイス部24から構成情報データおよび障害情報データを受取り、イベントリポジトリ28に保持させるためのイベントデータベース部26と、構成管理部30からPSマトリクスを受取って保持し、イベントデータベース部26から障害の症状イベントの集合を受けて、PSマトリクスと、実際に観測される症状イベントの集合とを比較し距離計算をして、障害の根本原因の問題を推論するための障害管理部34と、障害管理部34から障害の根本原因の問題の推論結果を示す情報を受け、ユーザに提示するためのユーザインターフェイス部36と、マスタNMSとワーカNMS間で要求や通知などを行うための関係NMSインターフェイス部38とを含む。

【0061】マスタNMS70、ワーカNMS72、74、76などによるPSマトリクスの生成およびイベント相関の処理については後述する。

【0062】図3に示されるネットワーク管理システムは、実際にはパーソナルコンピュータまたはワークステーションなど、コンピュータ上で実行されるソフトウェアにより実現される。図4に、ネットワーク管理システムを実現するコンピュータの外観を示す。図4を参照してこのコンピュータは、CD-ROM (Compact Disc Read-Only Memory) ドライブ50およびFD (Flexible Disk) ドライブ52を備えたコンピュータ本体40と、ディスプレイ42と、プリンタ44と、キーボード46と、マウス48とを含む。

【0063】図5に、このコンピュータの構成をブロック図形式で示す。図5に示されるようにこのネットワーク管理システム20を構成するコンピュータ本体40は、CD-ROMドライブ50およびFDドライブ52に加えて、それぞれバス66に接続されたCPU (Central Processing Unit) 56と、ROM (Read Only Memory) 58と、RAM (Random Access Memory) 60と、ハードディスク54とを含んでいる。CD-ROMドライブ50にはCD-ROM62が装着される。FDドライブ52にはFD64が装着される。

【0064】既に述べたようにこのネットワーク管理システムは、コンピュータハードウェアと、CPU56により実行されるソフトウェアとにより実現される。一般的にこうしたソフトウェアは、CD-ROM62、FD64などの記憶媒体に格納されて流通し、CD-ROMドライブ50またはFDドライブ52などにより記憶媒体から読取られてハードディスク54に一旦格納される。さらにハードディスク54からRAM60に読出されてCPU56により実行される。図4および図5に示したコンピュータのハードウェア自体は一般的なものである。したがって、本発明の最も本質的な部分はCD-ROM62、FD64、ハードディスク54などの記憶媒体に記憶されたソフトウェアである。

【0065】なお図4および図5に示したコンピュータ自体の動作は周知であるので、ここではその詳細な説明は繰返さない。

【0066】[システムの動作] 以下、図1～図5に示す本願発明の実施の形態に係るネットワーク管理システム20の動作について、さらに図6～図9をも参照して説明する。ネットワーク管理インターフェイス部24は、SNMP等のネットワーク管理プロトコルを用いて、管理対象装置のMIB等のデータを取得したりポーリングしたりする。またネットワーク管理インターフェイス部24は、管理対象装置からのTrapイベントを受信する。ネットワーク管理インターフェイス部24はさらに、管理対象装置から取得したデータのうち構成情報に関するデータは構成管理部30に、障害情報に関するデータはイベントデータベース部26に、それぞれ適当な形式に変換して送る。

【0067】構成管理部30は、図11および図13に示すような管理対象オブジェクトモデルと、これに関連した図15に示したようなイベント伝播モデルとを保持する。なおこれらモデルは、図5に示すハードディスク54内のファイル (図示せず) から読み込むこともできるし、ユーザインターフェイス部36を介してユーザが入力することも可能である。

【0068】構成管理部30はさらに、ネットワーク管理インターフェイス部24から図12に示されるネットワーク例のような構成情報データを受取り、管理対象オブジェクトモデルに基づいて図14に示されるような管

理対象オブジェクト構成情報を構築する。この管理対象オブジェクト構成情報は管理対象オブジェクトリポジトリ32に保持される。

【0069】構成管理部30はまた、自己の保持する管理対象オブジェクトモデル、イベント伝播モデル、および管理対象オブジェクトリポジトリ32の管理対象オブジェクト構成情報を用いて、図1に示すようなPSマトリクス90を生成し、障害管理部34に通知する。ワーカNMSでは、マスタNMSから関係NMSインターフェイス部38を通じて問題イベントリストが通知される。なお、問題イベントの指定はイベント伝播モデル記述時にも可能であるし、ユーザインターフェイス部36からユーザが症状イベントパターンマトリクスの生成を要求する際に、イベント伝播モデルを参照しながら指定することも可能である。

【0070】イベントデータベース部26は、構成情報データおよび障害情報データをイベントとしてネットワーク管理インターフェイス部24から受けて、それらをイベントリポジトリ28に保持する。イベントデータベース部26はこれらイベントの中で、障害情報に関するデータ、特に障害の症状イベントを障害管理部34に通知する。

【0071】障害管理部34は、構成管理部30からPSマトリクスを受け取り保持する。障害管理部34はまた、イベントデータベース部26から障害の症状イベントの集合（前述した、実際に観測される症状イベントパターン）を受取る。障害管理部34は、構成管理部30から受取ったPSマトリクスと、イベントデータベース部26から受取った実際に観測された症状イベントパターンとを比較して、障害の根本原因の問題の候補を特定する。特定された問題の候補はユーザインターフェイス部36に通知される。

【0072】ユーザインターフェイス部36は、図4および図5に示されるディスプレイ42などによりこの障害の根本原因の問題の候補をユーザに提示する。このとき、計算された相違度が最も小さい障害原因のみを提示してもよいし、相違度の小さいものから順に予め定められた個数だけ候補として提示するようにしてもよい。複数個提示する場合には、伝播モデルが完全でない場合、または観測した症状イベントにノイズ（発生するはずの症状イベントが観測されなかったり、関係のない症状イベントが観測されたりすること）が含まれる場合でも、候補リストの中に正しい解が含まれる可能性が高くなる。またこのリストの提示を受けて利用者が実際に候補としてあげられたネットワーク機器を調べることにより、最終的に原因を特定することが容易になる。

【0073】または、あるしきい値を設けておき、そのしきい値よりも小さな相違度を有する障害原因を表示するようにしてもよい。この方法は、経験的に、相違度がある値以下の障害原因の中に真の障害原因が含まれる可

能性が高いということが分かっている場合に、それらを全て利用者に通知するので、利用者は効率的に障害原因を探ることができて有利である。

【0074】[PSマトリクスの作成処理] PSマトリクス90の作成は以下のようにして行われる。なお、PSマトリクス90は、前述のように横方向の部分マトリクス92、94、96に分割される。このように分割された部分マトリクス92、94、96はそれぞれワーカNMS72、74、76が保持する。これは、イベント相関処理のために各NMSで貯えられた入力SymptomSetをNMS間で移動させる必要がなく、そのまま処理できるからである。

【0075】[マスタNMS70での処理] 図6を参照して、まずマスタNMS70で、利用者がサブスクライブしたいProblem リストを設定する。「サブスクライブ」とは、ある問題について通知を要求することをいう。マスタNMS70はそのProblem リストを各ドメインごとに分類し、ワーカNMS72、74、76に通知する（250）。ワーカNMS72、74、76は後述するように各々、この通知をトリガーとしてPSマトリクスの作成を開始する。PSマトリクスワーカNMS72、74、76はそれぞれ、図1に示した部分マトリクス92、94、96の作成を担当する。その処理の詳細については後述する。

【0076】例えばワーカNMS74は、第1のドメイン78のProblem リスト[P11, P12, ..., P1m1]の各々の問題に対して、まず第1のドメイン78のトポロジ情報（管理対象オブジェクト構成情報）およびイベント伝播モデルを用いて図1のマトリクスの[A1]の部分を作成していく。

【0077】ワーカNMS72はさらに、境界MOのSymptom に到達すると、マスタNMS70に対してPSマトリクス生成係属要求（境界MOを管理する他NMS名、ProblemID、境界MO名、Symptom名）を送信する。ここで、ProblemIDとは、MO名+Problem名で一意に特定されるProblem識別子である。同様に、SymptomIDは、MO名+Symptom名で一意に特定されるSymptom識別子である。MO名はドメイン内で一意になるように、構成管理部30で管理されており、MO名からその管理対象オブジェクトのクラス（MOクラス）を識別できる。MOクラスが特定できると、イベント伝播モデルすなわち各Symptomのイベント伝播が分かるものとする。

【0078】再び図6を参照して、マスタNMS70は、この要求があるかないかを判定し（252）、ある場合にはこれに回答して、境界MOを管理する他NMSに対してPSマトリクス生成継続要求（ProblemID、境界MO名、Symptom名）を送信する（254）。

【0079】ワーカNMS74、76ではそれぞれ、この要求メッセージをトリガーとして、自分の担当するド

メインのトポロジー情報を用いて図1の[B1]、[C1]を埋めていく。ワーカNMS72は逆に、他のワーカNMS74、76のProblemからスタートして、自己のドメインのトポロジー情報を用いて図1の[A2]、[A3]を埋めていく。こうした処理をワーカNMS72、74、76の間で相互に行うことにより図1のPSマトリクス90を構成する部分マトリクス92、94、96がそれぞれ形成される。

【0080】ワーカNMS72、74、76はそれぞれ、あるProblemについてPSマトリクスの生成処理を進める。この場合、そのProblemは各自の担当するドメインのProblemとは限らない。ワーカNMS72、74、76は、あるProblemに対する全てのイベント伝播が終了し末端のSymptomまで到達すると、次のProblemの処理に移る。処理すべき全てのProblemのPSマトリクス生成が終了すると、マスタNMS70に対して自己の担当分のPSマトリクス生成が終了したことを通知する。ワーカNMS72、74、76は、一度自己のPSマトリクス生成が終了しても、PSマトリクス生成継続要求を受信すると、またPSマトリクス生成処理を開始し、終了すると前述したのと同様にマスタNMS70に対して自己の担当分のPSマトリクス生成が終了したことを通知する。

【0081】マスタNMS70は、Problemリストを通知した後または最後に生成継続要求を送信した後に全てのワーカNMSからPSマトリクス生成の終了通知を受信すると、PSマトリクス生成処理が完了したものと判断し(256)、観測SymptomSet相関処理の開始を全てのNMSに対して要求する。

【0082】具体的に図7を参照して上述の動作を説明すると、まずマスタNMS70に対して、利用者がサブスクライブを希望するProblemリストを設定する(300)。これに回答してマスタNMS70が各ワーカNMSに対してProblemリストを通知する(302)。各ワーカNMSがこれに対してPSマトリクス生成処理を開始する。

【0083】図7に示す例では、まず第1のワーカNMS72がPSマトリクス生成処理を終了し終了通知をマスタNMS70に対して送信する(304)。続いてワーカNMS74がワーカNMS76に対するPSマトリクス生成継続要求をマスタNMS70に対して送信(306)した後終了通知を送信する(308)。マスタNMS70はこのPSマトリクス生成継続要求306に回答してワーカNMS76に対してPSマトリクス生成継続要求を送信する(310)。ワーカNMS76はPSマトリクスの生成処理の実行を継続するが、ワーカNMS74に対するPSマトリクス生成継続要求312をマスタNMS70に対して送信した後、終了通知を送信する(314)。マスタNMS70はこのPSマトリクス生成継続要求312に回答してワーカNMS74に対し

てPSマトリクス生成継続要求316を送信し、ワーカNMS74はこの要求に回答して、一旦終了したPSマトリクス生成処理を再開し、処理が終了すると終了通知318をマスタNMS70に対して送信する。こうして、マスタNMS70は全てのワーカNMSから終了通知を受取るとPSマトリクスの生成処理が完了したと判断し(320)、次の観測SymptomSet相関処理を開始する。

【0084】[ワーカNMSでのPSマトリクス作成処理] 各ワーカNMSでのPSマトリクス作成処理について図8を参照して説明する。まず、マスタNMS70からのProblemリスト通知またはPSマトリクス生成継続要求を受信すると、未処理ProblemキューにProblemDataとしてProblemリストを入れる(260)。各ProblemDataは情報としてProblemIDと、境界MOのMO名であるInitialMOと、Symptom名の初期値であるInitialSymptomを持つ。ProblemIDは、前述のとおりMO名とProblem名とからなる。

【0085】続いて、未処理Problemキューが空か否かを判定する(262)。空であれば処理終了である。空でなければ未処理ProblemキューからひとつだけProblemDataを取出し、SymptomDataとして未処理Symptomキューに入れる(264)。SymptomDataは情報としてProblemIDとSymptomID(MO名とSymptom名)とを持つ。この処理ではProblemDataのProblemIDをSymptomDataのProblemIDに代入する。またProblemDataのInitialMOをSymptomIDのMO名に代入する。ProblemDataのInitialSymptomをSymptomIDのSymptom名に代入する。

【0086】未処理Symptomキューが空か否かを判定する(266)。空であれば制御はステップ262に戻る。空でなければ、未処理Symptomキューから1つだけSymptomDataを取出す(268)。このSymptomDataに対して、そのMOと接続関係のあるMOが存在し、かつそのMOのクラスにSymptomDataのSymptomからのイベント伝播が定義されているか否かを判定する(270)。定義されていなければ制御はステップ266に戻る。

【0087】定義されていれば、新しいSymptomDataを以下のようにして生成する(272)。すなわち、MO名にはその接続関係のあるMO名を、Symptom名にはイベント伝播先のSymptomを、それぞれ代入する。こうして生成されたSymptomDataを未処理symptomキューに入れる。また、PSマトリクスの(ProblemID, SymptomID)に対応する欄に「1」を設定する。

【0088】次に、SymptomDataのMOが境界MOか否かについての判定を行い(274)、境界MOであればマスタNMS70に対してPSマトリクス生成継続要求を送信して(276)制御をステップ270に戻す。境界MOでなければ何もせず制御をステップ270に戻す。

す。

【0089】こうした処理によって、各ワーカNMSでは各自の担当する部分PSマトリクスが生成される。

【0090】〔観測SymptomSet相関処理〕この処理は非常に簡単である。前述のとおりマスタワーカモデルで処理する。入力SymptomSet[S11', S12', ..., S1n1', S21', S22', ..., S2n2', S31', S32', ..., S3n3'] は、以下のようにわけられる。

【0091】

- ・入力SymptomSet1[S11', S12', ..., S1n1']
- ・入力SymptomSet2[S21', S22', ..., S2n2']
- ・入力SymptomSet3[S31', S32', ..., S3n3']

これらはもともとワーカNMS 72、74、76で分散管理されている。

【0092】そして各ワーカNMSは、自己の保持する部分PSマトリクスと自己に入力されるSmptomSet とを用いて相関値（例えば距離）を計算する。たとえばワーカNMS 72は、自己の保持する部分マトリクス92と、自己に入力されるSmptomSet1とを用いて相関値を計算する。そして、全てのProblem（または上位の候補のProblemのみ）の相関値をマスタNMSに通知する。

【0093】マスタNMS 70は、全てのワーカNMSから結果を受取ると、各Problemごとに相関値（距離）を集計して最大相関値（最小距離）を持つProblemを求める。または、上位のいくつかのProblemを候補として選ぶ。または、所定のしきい値をこえる相関値（または所定のしきい値より小さな距離）を持つProblemを候補として選ぶ。

【0094】この処理におけるマスタNMS 70と各ワーカNMS 72、74、76との間のインタラクションの例について図9を参照して説明する。まず、PSマトリクスの生成処理が完了すると（320）、マスタNMS 70は各ワーカNMS 72、74、76に対して観測SymptomSet相関処理の開始要求を送信する（280）。各ワーカNMS 74、76、78からは、処理が終了次第順次終了通知がマスタNMS 70に対して送信されてくる（282、284、286）。全ての終了通知を受信すると、マスタNMS 70は前述した集計処理を行い、候補のProblemを求めてユーザインターフェイス部36に与える（288）。

【0095】こうして、分散環境での観測SymptomSet相関処理を効率良く行うことができる。この場合、従来のように一箇所でPSマトリクスを保持する方式ではないので、リソースの制限に対してもPSマトリクスを分散保持することで対応できるという効果がある。また、従来のように各ドメイン別に個別にPSマトリクスを生成保持して観測SymptomSet相関処理を行う場合と比較して欠落する情報が少なく、ネットワークが大規模になっても正確な問題特定を行うことができるという効果がある。

【0096】なお、この実施の形態のシステムでは、PSマトリクスの生成から各ワーカNMSで分散して行っている。しかし本発明はこうした方式に限定されるわけではなく、PSマトリクスの生成はリソースに余裕のあるマスタNMSで行い、出来上がったPSマトリクスを分割してそれぞれの部分相関マトリクスを各ワーカNMSに送信するようにしてもよい。特にこの方式は、特に、PSマトリクス生成のための各NMS間のインタラクションが多くなる場合であって、かつPSマトリクスがそれほど大きくない場合に効果的である。なお、境界MOの数が多くなればインタラクションはそれほど発生しないと思われる。

【0097】一箇所でPSマトリクスを生成する場合には、必要な情報は以下のとおりである。

【0098】・イベント伝播モデル（簡単のため、クロスドメインで共通とする。）

- ・クロスドメインのトポロジー情報
- ・サブスクライブするProblem リスト

この場合には、ネットワークが大きくなるとトポロジー情報が膨大になる可能性があるが、PSマトリクス生成に真に必要な最小限のトポロジー情報のみに絞り込めば情報量をかなり削減することが可能である。具体的には、moName（MOの名称）、moClassName（MOのクラス名）、relationship（関係：関係名と接続MO名）リスト程度で十分であり、構成管理の属性等は不要である。

【0099】＜第2の実施の形態＞なお、図10に示すように、ネットワーク管理システム20を、既存のネットワーク管理システム110を介して管理対象ネットワーク22に接続してもよい。

【0100】以上、本願発明にかかるネットワーク管理システムを実施の形態に基づいて説明してきたが、本願発明はこれら実施の形態のシステムに限定されるわけではない。本願発明の権利範囲は、特許請求の範囲の各請求項の記載によって定められるべきである。本願明細書に開示された実施の形態の各構成要素と均等の構成要素を用いたものも本願発明の権利範囲に含まれる。

【図面の簡単な説明】

【図1】図1は、本願発明で用いられるPSマトリクスの例を示す図である。

【図2】図2は、本願発明を実施するためのマスタ・ワーカ・モデルを示す図である。

【図3】図3は、本願発明の一実施の形態に係るネットワーク管理システムのブロック図である。

【図4】図4は、本願発明に係るネットワーク管理システムを実現するためのコンピュータの外観図である。

【図5】図5は、図4に示されるコンピュータのブロック図である。

【図6】図6は、本願発明の一実施の形態のネットワーク管理システムのマスタNMS 70で行われる処理のフ

ローチャートである。

【図7】図7は、PSマトリクス生成処理におけるマスタNMSおよびワーカNMS間のインタラクションを示す図である。

【図8】図8は、本願発明の一実施の形態のネットワーク管理システムの各ワーカNMSで行われるPSマトリクス作成処理のフローチャートである。

【図9】図9は、観測SymptomSet相関処理におけるマスタNMSおよびワーカNMS間のインタラクションを示す図である。

【図10】図10は、本願発明の他の実施の形態にかかるネットワーク管理システムを示すブロック図である。

【図11】ネットワーク管理に使用される管理対象オブジェクトモデルのクラス図である。

【図12】図12は、簡単なネットワークを模式的に示す図である。

【図13】図13は、図12に示されるネットワークの管理対象オブジェクトモデルのクラス図である。

【図14】図14は、図12に示されるネットワークの管理対象オブジェクトを示すインスタンス図である。

【図15】図15は、ネットワーク管理で用いられるイベント伝播モデルのクラス図である。

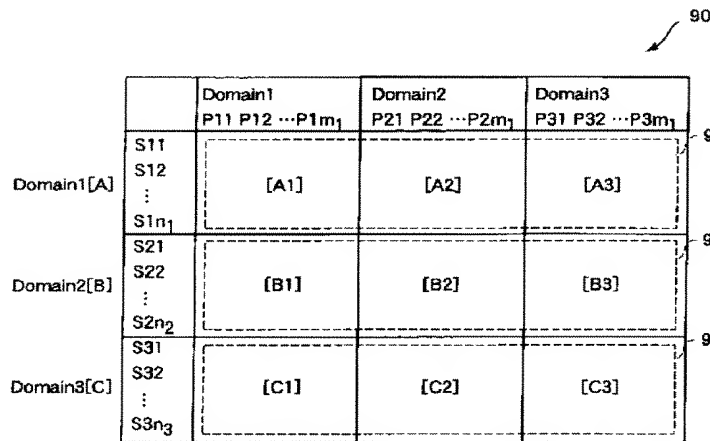
【図16】図16は、従来の症状イベント伝播モデルに基づく症状イベントパターンを表形式で示す図である。

【図17】図17は、従来技術の分散環境での、各ドメインで使用されるPSマトリクスを示す図である。

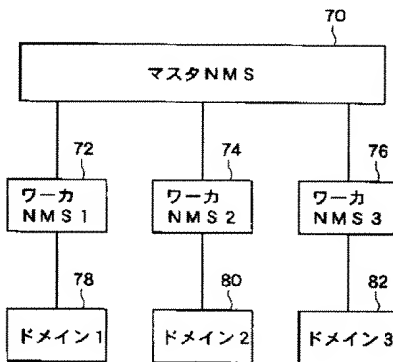
【符号の説明】

- 20 ネットワーク管理システム
- 22 管理対象ネットワーク
- 24 ネットワーク管理インタフェース部
- 26 イベントデータベース部
- 28 イベントリポジトリ
- 30 構成管理部
- 32 管理対象オブジェクトリポジトリ
- 34 障害管理部
- 36 ユーザインタフェース部

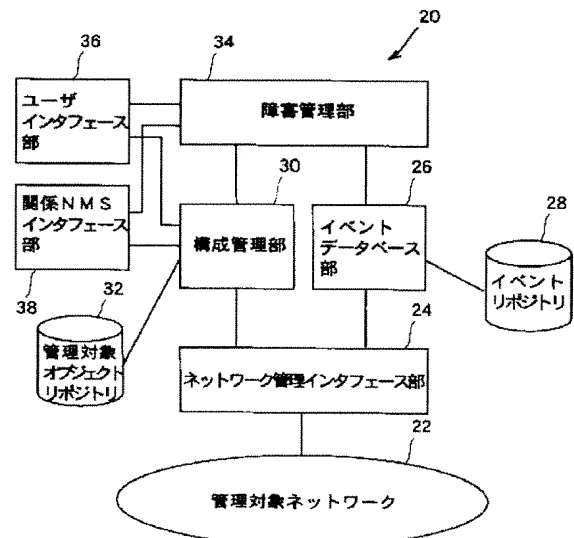
【図1】



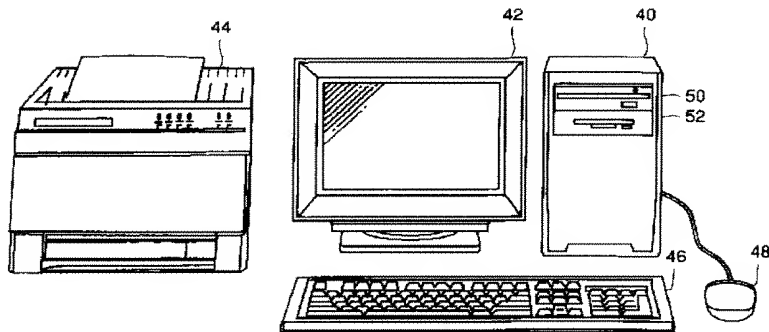
【図2】



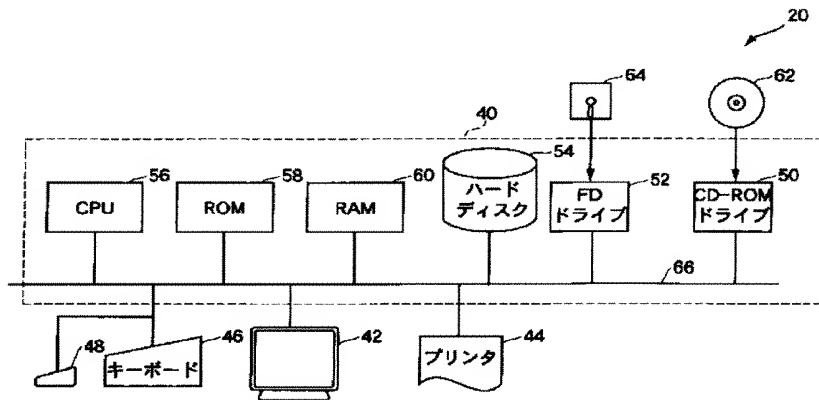
【図3】



【図 4】



【図 5】



【図 17】

| Domain1[A]       |  |
|------------------|--|
|                  | Domain1<br>P11 P12 ...P1m <sub>1</sub> |
| S11              |  |
| S12              |  |
| ⋮                |  |
| S1n <sub>1</sub> |  |

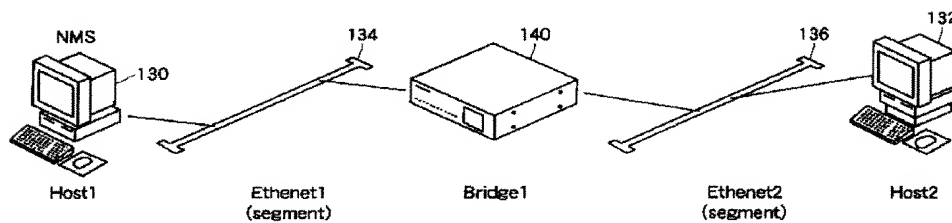
Domain2[B]

| Domain2          |                             |
|------------------|-----------------------------|
|                  | P21 P22 ...P2m <sub>1</sub> |
| S21              |                             |
| S22              |                             |
| ⋮                |                             |
| S2n <sub>2</sub> |                             |

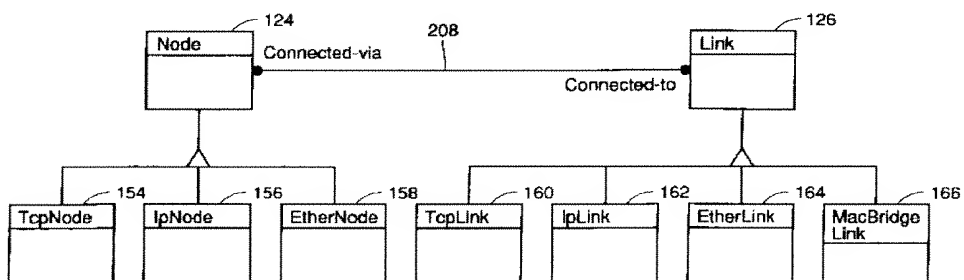
Domain3[C]

| Domain3          |                             |
|------------------|-----------------------------|
|                  | P31 P32 ...P3m <sub>1</sub> |
| S31              |                             |
| S32              |                             |
| ⋮                |                             |
| S3n <sub>3</sub> |                             |

【図 12】

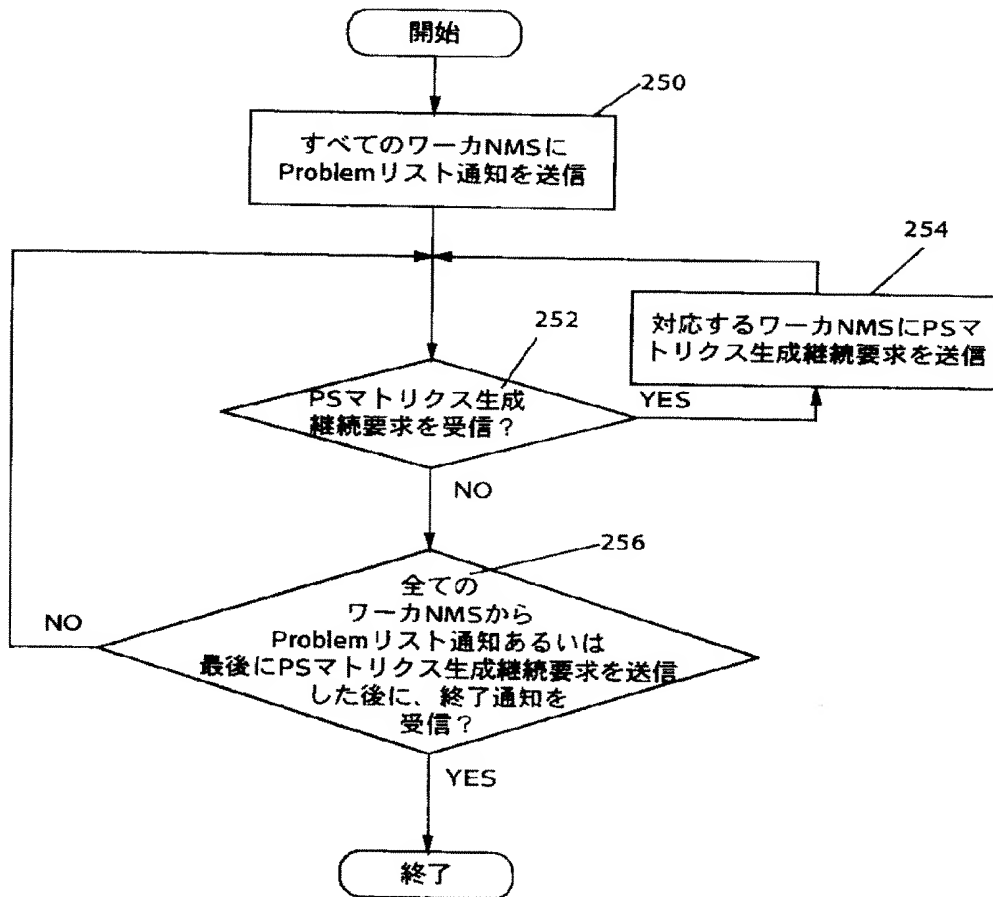


【図 13】

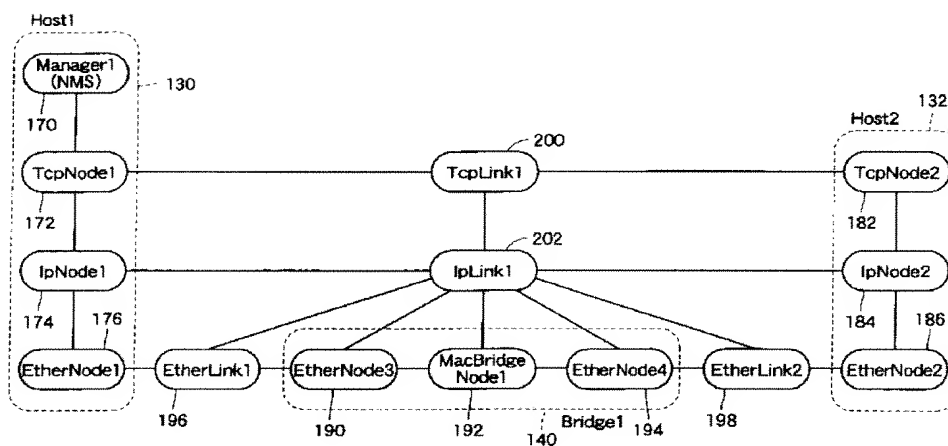




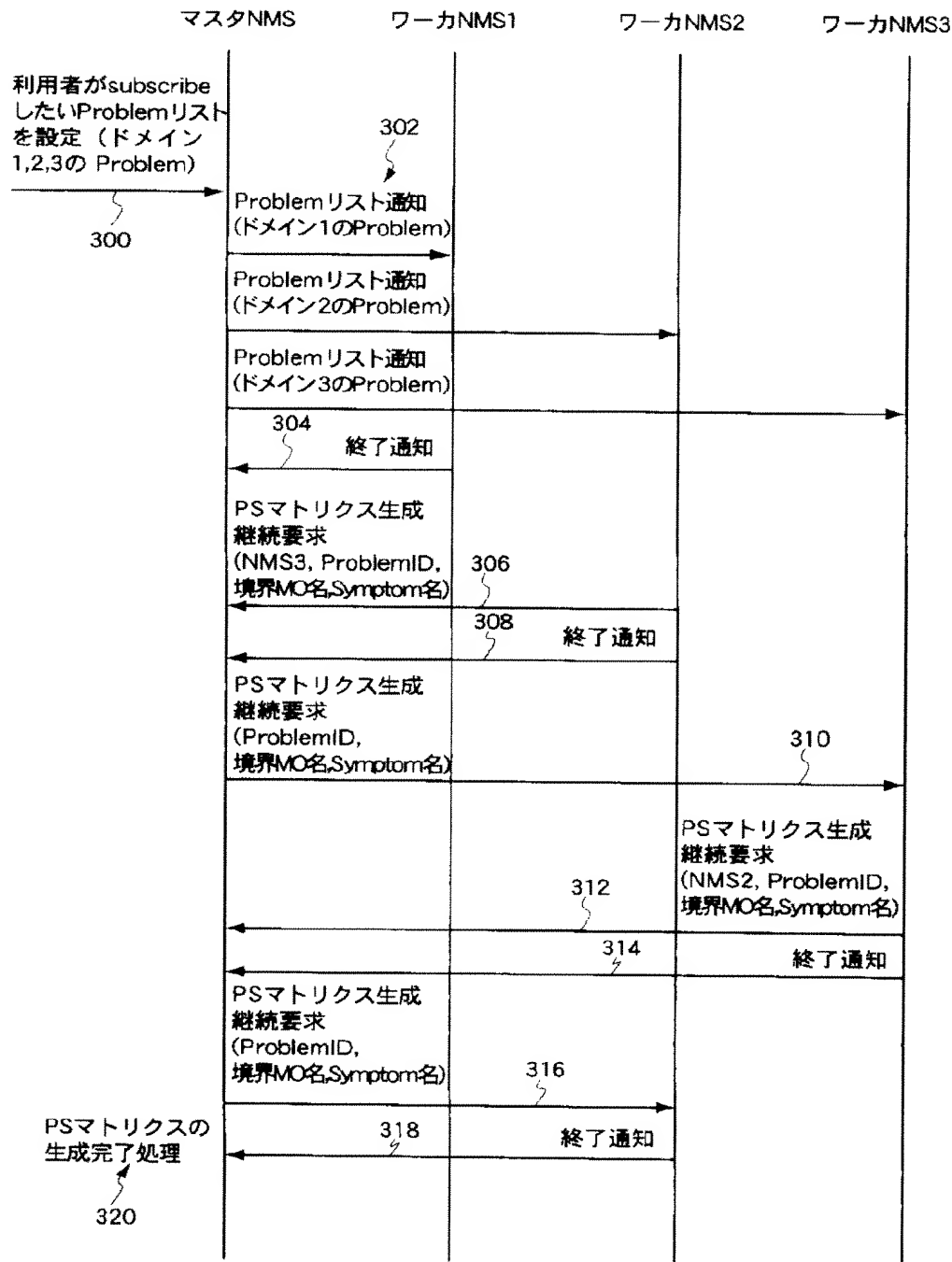
【図6】



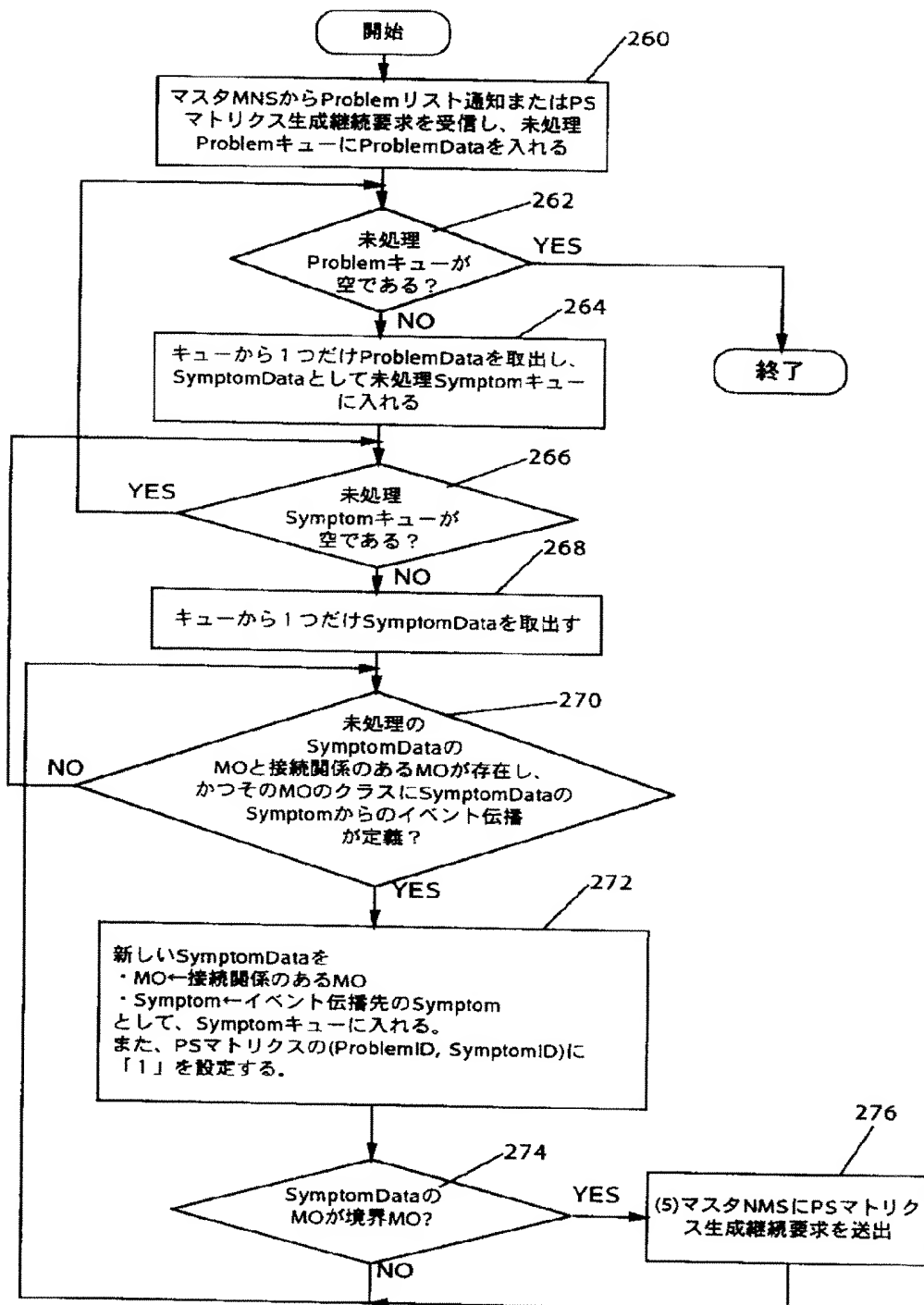
【図14】



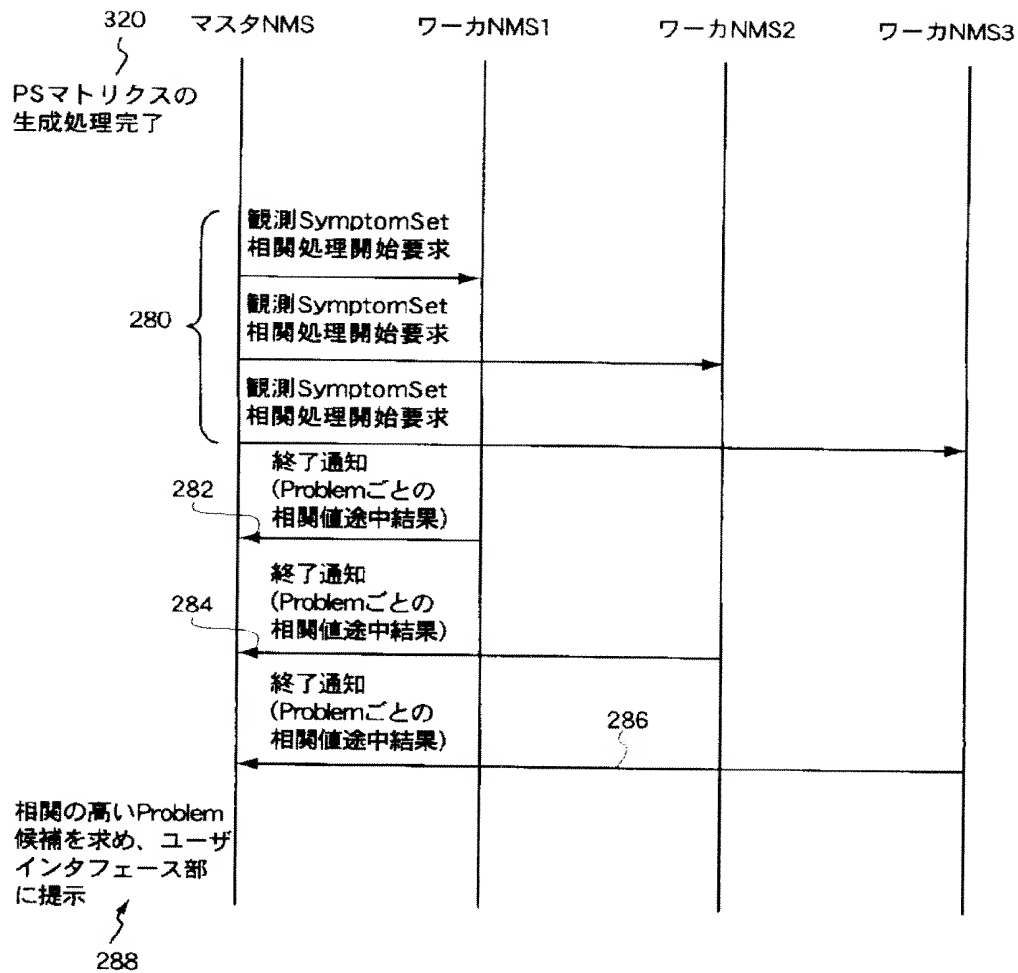
【図7】



【図8】



【図9】

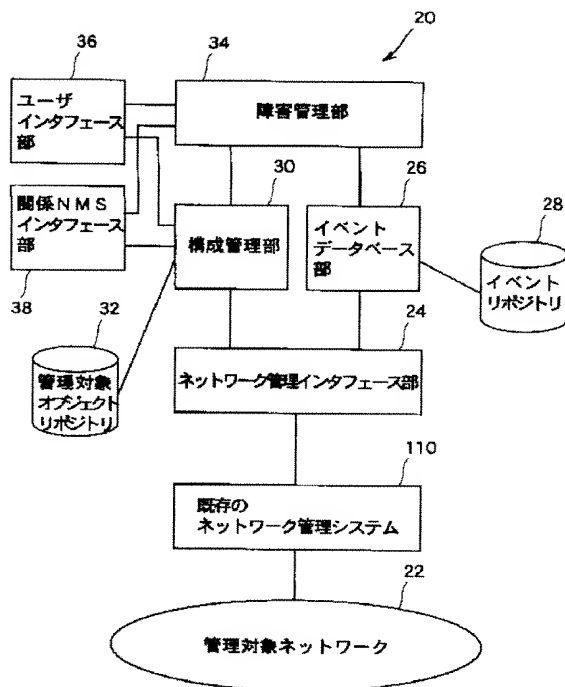


【図16】

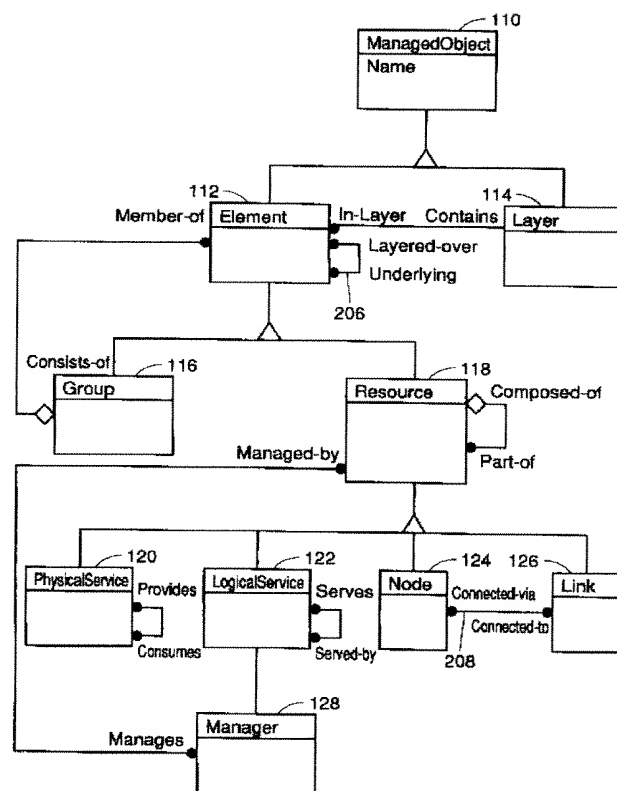
イベント伝播モデルに基づく症状イベントパターン

| 問題イベント<br>症状イベント              | ケース1：<br>EtherLink1.Down | ケース2：<br>EtherLink2.Down |
|-------------------------------|--------------------------|--------------------------|
| EtherNode1.Down               | 1                        |                          |
| TcpNode1.Disconnect           | 1                        | 1                        |
| EtherNode2.Down               |                          | 1                        |
| TcpNode2.Disconnect           | 1                        | 1                        |
| EtherNode3.Down               | 1                        |                          |
| EtherNode4.Down               |                          | 1                        |
| MacBridgeNode1.CannotBridging | 1                        | 1                        |
| EtherLink1.Down               | 1                        |                          |
| EtherLink2.Down               |                          | 1                        |
| TcpLink1.NotResponding        | 1                        | 1                        |

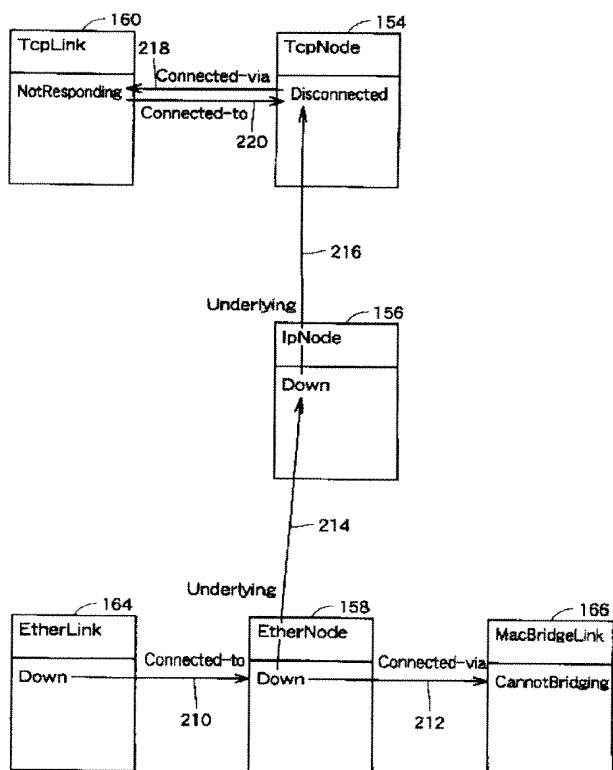
【図 10】



【図 11】



【図 15】



フロントページの続き

(51) Int. Cl. 7

識別記号

F I

テームコード (参考)

H O 4 L 29/14

F ターム (参考) 5B045 BB47 JJ02 JJ08 JJ09 JJ14  
5B089 AA03 AA16 AC03 CC15 DD02  
EA10  
5K030 GA12 LE01 MA01 MB20 MD00  
5K033 AA06 BA04 DA01 EA04 EA07  
EC00  
5K035 AA03 AA07 DD01 JJ01 JJ03